

清 华 大 学

# 综合论文训练

题目：基于深度学习的多摄像头行人位置估计算法研究

系 别：自动化系

专 业：自动化系

姓 名：雷世龙

指导教师：冯建江 副教授

2022 年 5 月 29 日



# 关于学位论文使用授权的说明

本人完全了解清华大学有关保留、使用学位论文的规定，即：学校有权保留学位论文的复印件，允许该论文被查阅和借阅；学校可以公布该论文的全部或部分内容，可以采用影印、缩印或其他复制手段保存该论文。

(涉密的学位论文在解密后应遵守此规定)

签 名：\_\_\_\_\_ 导师签名：\_\_\_\_\_ 日 期：\_\_\_\_\_



## 中文摘要

在机器学习技术迅速发展的背景下，多摄像头行人位置估计因为其在智能城市、视频监控、自动驾驶和机器人等领域中广泛的应用需求而受到了越来越多的重视。随着科技的进步和社会的发展，多摄像头行人位置估计技术的应用需求也越来越大，例如，在汽车自动驾驶技术的实现中需要判断视线中人物的位置，在多个视频视角重叠监控的路口需要判断行人穿行马路时的位置。然而，目前的多摄像头行人位置估计技术仍然有很大进步空间，如在行人数量较多、多个行人互相遮挡或有车辆遮挡视角时，算法效果仍难以达到预期。如何在提高估计准确度的同时提升鲁棒性，这个问题具有相当的挑战性。近年来，深度学习方法在各行各业尤其是计算机视觉领域迅速发展，在多个领域例如行人跟踪与识别等的应用极大推动了多摄像头行人位置估计技术的成熟，行人位置估计算法有了新的突破口。因此，本文将传统方法与深度学习方法相结合，实现了从摄像头数据得到行人位置估计的过程，探索更具有鲁棒性的行人位置估计算法。

本文提出了一种结合各摄像头视角信息对行人位置进行高斯概率分布建模，然后利用深度学习网络训练实现行人位置估计的算法。本文提出的算法首先利用即插即用的单目检测模型得到每个摄像头中每个行人的姿态节点，然后利用相机投影的几何关系，投影得到行人在实际路面上的位置。之后，根据摄像头视角，在行人所在位置建立二维高斯概率分布，作为行人位置的估计。而后利用高斯概率分布之间的 KL 散度作为距离进行层次聚类，从而融合每个摄像头对单个行人的位置估计。得到最佳参数后，将每一帧得到的高斯概率分布制作成热图，输入 ResUnet++ 深度学习网络进行图像分割的训练，最终得到行人位置估计结果。实验结果表明，在 WildTrack 数据集上，本文算法提高了不同摄像头视角信息的利用率，增强了算法鲁棒性，同时，算法效果接近业界领先水平。

**关键词：**多视角行人检测；深度学习算法；概率分布建模



## ABSTRACT

With the rapid development of machine learning technology, multi-view pedestrian position estimation has attracted more and more attention because of its wide application needs in intelligent cities, video surveillance, autonomous driving, robots and other fields. With the progress of science and technology and the development of society, there is an increasing demand for multi-view pedestrian position estimation technology. For example, it is necessary to estimate the position of people in the sight of auto-driving vehicles, and it is necessary to estimate the position of pedestrians when crossing the road at the crossroads where multiple video perspectives overlap. However, there are still great prospects for development in the field of current multi-view pedestrian position estimation technology. For example, when there are a large number of pedestrians, several pedestrians would block each other, or vehicles would block the view. In these circumstances, it is hard for current algorithms produce a marked effect. How to improve the estimation accuracy and robustness at the same time is quite challenging. In recent years, the deep learning method has developed rapidly in all professions and trades, especially in the field of computer vision. The application in many fields, such as pedestrian tracking and recognition, has greatly promoted the development and maturity of multi-view pedestrian position estimation technology. These new advances motivated the pedestrian position estimation algorithm to make a new breakthrough. In this context, this thesis combines the traditional method with the deep learning method to reproduce the process of pedestrian position estimation from the camera data, and explore a more robust pedestrian position estimation algorithm. In this thesis, we propose estimate the pedestrian location by combining the Gaussian probability distribution from different cameras and angles, and then performing training on a deep learning network. The algorithm proposed firstly uses an off-the-shelf monocular detection model to get the pose joints of each pedestrian in each camera, and then uses the geometric relationship of camera projection to get the location of the pedestrian on the ground plane. Then, depending on the location of each pedestrian and the angle of each view, a two-dimensional Gaussian probability distribution is established at the pedestrian location. Then the KL divergence between Gaussian probability distributions is used for hierar-

chical clustering, so as to fuse each camera information to estimate the only one position of a single pedestrian. After optimal parameters are obtained, the Gaussian probability distribution established in each frame is made into heatmaps, which will be input into the ResUNet++ for image segmentation training. Finally, the pedestrian position estimation results can be obtained. The experiments performed on the Wildtrack dataset. The results show that this algorithm makes better use of camera information from different views and enhances the robustness of prediction. Furthermore, the effect of the proposed algorithm is close to the state-of-the-art level.

**Keywords:** Multi-view Pedestrian Detection; Deep Learning Algorithm; Probability Distribution Modeling

# 目 录

<b>第 1 章 引言 .....</b>	<b>1</b>
1.1 问题描述 .....	1
1.2 课题研究背景及意义 .....	1
1.3 研究背景与现状 .....	2
1.3.1 单目行人检测算法 .....	2
1.3.2 行人重识别算法 .....	3
1.3.3 多摄像头行人位置估计算法 .....	5
1.4 论文结构安排 .....	6
<b>第 2 章 行人检测与高斯概率分布建模 .....</b>	<b>9</b>
2.1 数据集扩充和预处理 .....	9
2.2 行人检测算法 .....	11
2.3 相机投影原理 .....	13
2.4 高斯概率分布建模 .....	15
2.5 本章小结 .....	17
<b>第 3 章 数据准备与深度学习方法构建 .....</b>	<b>18</b>
3.1 层次聚类方法 .....	18
3.2 深度学习数据准备 .....	20
3.3 深度学习神经网络 .....	22
3.3.1 神经网络选择 .....	22
3.3.2 神经网络结构 .....	22
3.3.3 损失函数 .....	25
3.4 本章小结 .....	25
<b>第 4 章 评价指标与实验结果 .....</b>	<b>26</b>
4.1 评价指标 .....	26
4.2 实验设置和结果分析 .....	27
4.3 本章小结 .....	28

第 5 章 结论与展望 .....	29
5.1 结论 .....	29
5.2 展望 .....	29
参考文献 .....	31
致    谢 .....	35
声    明 .....	37

## 主要符号表

HOG	梯度方向直方图 (Histogram of Oriented Gradient)
SVM	支持向量机 (Support Vector Machine)
Re-ID	行人重识别 (Re Identification)
LBP	局部二值模式 (Local Binary Patterns)
CNN	卷积神经网络 (Convolutional Neural Networks)
RNN	循环神经网络 (Recurrent Neural Networks)
AMOC	累计运动背景网络 (Accumulative Motion Context Network)
ROI	感兴趣区域 (Region of Interest)
POM	概率图占据方法 (Probabilistic Occupancy Map method)
CRF	条件随机场 (Conditional Random Field)
ASPP	空洞空间金字塔池化 (Atrous Spatial Pyramid Pooling)



# 第1章 引言

## 1.1 问题描述

行人位置估计是计算机视觉中的经典问题，也是长期以来难以解决的问题。和人脸检测问题相比，由于人体的姿态复杂，变形更大，附着物和遮挡等问题更严重，因此准确的检测处于各种场景下的行人具有很大的难度。位置估计要解决的问题是：找出图像或视频帧中所有的行人，并估计出每个行人的3D真实位置。实际上，从要解决的问题上来看，行人位置估计相当于跨摄像头目标跟踪和多目标跟踪的融合，从所需技术上看，其包含行人检测、相机投影、数据融合、行人重识别等多个领域。

## 1.2 课题研究背景及意义

近年来，社会科技水平迅速发展，机器学习与人工智能领域在实际社会生活中也得到了越来越广泛的应用。2017年国务院发布的《新一代人工智能发展规划》中指出：人工智能将成为国与国之间新的竞争领域，人工智能是能够引领未来，颠覆未来的国家战略性技术，我国需要将人工智能技术作为提高国家竞争力，保障国家安全的重大战略。同时，由于社会信息化水平的迅猛发展，我国近年来监控摄像头在各个公共场所得到迅速普及。在学校、医院、车站、银行甚至是每个人的家中，监控摄像头的身影无处不在，各种摄像头产生了海量的视频数据，因此，如何利用这些视频数据便成为一个具有相当广阔前景的研究方向。例如，如今新型冠状病毒肆虐全球，如何利用无处不在的摄像头判断出行人体温，并对其位置加以追踪？又如当今自动驾驶技术飞速发展，如何利用汽车前置摄像头判断出行人位置，从而确定规避策略？这种背景凸显了计算机视觉技术的关键性。而在计算机视觉领域，机器学习与人工智能技术的广泛应用取得了相当好的效果。在深度学习方法下，以往在目标跟踪、轨迹检测、视频分类、行为分析、动作识别、自动驾驶等领域棘手的课题也取得了突破性的发展。行人位置估计作为计算机视觉领域一个关键并具有挑战性的问题也不例外。同时，由于视频序列并非互相独立，而是以时间维度的连续序列，因此在时间的维度上存在动作的连续性，若对此特点加以挖掘，可大大提升深度学习方法在视频序列分析上的效果。尤其是在多摄像头行人位置估计领域，深度学习方法仍然有很大的探索空间和应用价

值。因此，多摄像头行人位置估计正在受到越来越多的关注和重视。目前，多摄像头行人估计是很多热门技术的基础，在智能监控、客流量统计和自动驾驶方面均有应用。

在智能监控和轨迹预测领域，多视角行人位置估计可获得行人和车辆的行为轨迹，也可以对其未来轨迹进行预测，从而判断那些人物与车辆有意外的危险，提前发出警告或做好记录，也可以根据行人或车辆的预测轨迹进行警示，以阻止违反交通规则的行为。

在自动驾驶领域，多视角行人位置估计可利用汽车前置摄像头以及周边环境监控摄像头预测交通事故的发生可能性，从而帮助汽车智能选择行进决策、安排行驶路线、协调速度和方向。

在一般场景下，目前已经存在较高精度的多视角行人位置估计算法。但在较为密集的场景下，目前的算法鲁棒性一般，这是因为较为密集的场景下存在着比较严重的互相遮挡问题，从而造成对行人特征提取不全、信息缺时的问题。同时，不同摄像头视角的信息也没有得到充分的利用。可见，应对密集场景下的行人位置估计，提高算法鲁棒性具有着重要的研究意义。

### 1.3 研究背景与现状

多摄像头行人位置估计问题具有相当多的技术难点，例如行人外观差异大、密集行人的互相遮挡、背景复杂难以分割、多视角数据融合和行人的重识别、模型复杂检测速度底下等。早期行人位置估计多使用图像处理以及模式识别中的传统方法，准确率较低，效果欠佳。然而随着深度学习的迅速发展，行人检测和重识别在性能上有了巨大的突破。一方面，越来越多的数据集被发布，例如 WildTrack<sup>[1]</sup>数据集，提供了大量的训练样本，促进了深度学习在此领域的应用；另一方面，用于不同研究目的的不同结构的深度学习网络也相继涌现，网络的深度也大幅深入，大大提高了网络的学习能力和预测精度。下文将对近年来推动多摄像头行人位置估计算法的技术进行介绍。

#### 1.3.1 单目行人检测算法

单目行人检测的任务是通过单个摄像头采集的数据，找出视频帧中的行人，也可显示其位置。检测结果一般用图像中的矩形框来表示。单目行人检测可分为基于运动检测的算法和基于机器学习的算法。

基于运动检测的算法基本原理是先对视频做背景建模，再利用获取到的背景

参考与当前图像序列的当前帧做差分，从而计算出背景图像像素差异，设定阈值将运动目标和背景进行分割。基于运动检测的经典算法包括单高斯算法、帧间差算法、高斯混合模型分离算法<sup>[2]</sup>、VIBE 算法<sup>[3]</sup>、CodeBook 算法<sup>[4]</sup>等，这些传统方法各有特点和优势，但主要流程都是先通过背景建模提取得到视频中的运动目标，再通过分类器来判断出行人。其主要优势在于算法简单、运行比较高效，甚至可以达到实时监测的效果。但这些方法也存在一些缺陷，例如需要摄像头固定，且很容易受到环境和背景噪声的影响，这对环境和设备有很大的要求，并且产生了很多不确定因素。另外，这些方法的一大缺点是只能检测运动目标而不能检测静止目标，这大大增加了算法的局限性。

基于机器学习的算法，基本原理是基于 HOG 的特征提取和 SVM 分类器的行人检测。HOG(Histogram of Oriented Gradient)<sup>[5]</sup>特征是基于本地像素块进行特征直方图提取的一种算法。而 SVM(Support Vector Machine) 的原理则是通过找到一个超平面使得超平面离最近的样本点距离最大化，以对样本进行分类。然而，HOG 和 SVM 相结合的方法有一个比较大的缺陷就是计算量较大，计算速度比较慢，此外，其对遮挡的鲁棒性很差。此后，积分通道特征 (integral channel features)<sup>[6]</sup> 和部分检测的方法也被提出，以对上述方法进行优化。

近年来，由于深度学习方法的迅速发展和应用，单目行人检测算法也迎来了新的研究热潮。MonoPair<sup>[7]</sup>使用经过训练的网络来获取检测到的人或其他目标的 3D 检测框 (bounding box)，然后添加成对的空间关系 (基于与目标之间中点相关的预测约束)，以改进对行人位置估计的结果。MonoLoco<sup>[8]</sup>在单目行人检测领域提出基于不确定性估计来推断每个行人的 2D 检测框的深度。它采用 2D 估计来建模模糊的 3D 位置，而这种模糊特征与被跟踪人群的内在特征有关。Hayakawa 和 Dariush<sup>[9]</sup>通过引入不对称损失函数改进了 MonoLoco 的 3D 位置估计方法。它可以更好地处理了远处行人估计关节的像素相关误差，从而提高了位置估计精度。

### 1.3.2 行人重识别算法

行人重识别 (Person Re-identification, Re-ID) 的目标是利用计算机视觉技术在已有的可能来源与非重叠摄像机视域的视频序列中识别出目标行人。在监控视频中，由于相机分辨率和拍摄角度等各种外在条件的限制，往往难以得到目标人物较为清晰的脸部图像，因此人脸识别失效。这种情景就是行人重识别就成为一个重要应用场景。当前行人重识别的研究主要分为两大类：closed-world 和 open-world。closed-world 主要是从行人的检测框图片中去检索目标行人，而

open-world 重在直接从视频中去检索目标行人，或者是偏向无监督、弱监督学习。目前，机器学习方法在行人重识别算法中得到了广泛应用，其主要包括基于表征学习的 Re-ID 方法、基于度量学习的 Re-ID 方法、基于局部特征的 Re-ID 方法和基于视频序列的 Re-ID 方法。

表征学习 (Representation learning) 方法在行人重识别领域的应用很常见。得益于卷积神经网络 (Convolutional neural network, CNN) 的迅速发展，从图像中根据人物需求提取表征特征的技术越来越成熟。基于特征表示的方法重点在于在光照和视角变化的情况下提高行人图像特征表示模型的鲁棒性。其提取的特征主要有底层视觉特征、中层语义特征和高级视觉特征三个方面。对于底层视觉特征，这种方法主要将图像划分为多个区域，对每个区域底层视觉特征提取后进行组合。由于多数情况下行人衣着颜色比较简单，因此最常用的是 RGB、HSV 颜色直方图，这是因为颜色特征在不同光照或角度等行人识别的不适环境中具有一定的不变性。此外，还有 Haar-like Representation、局部二值模式 (LBP)、Gabor 滤波器、共生矩阵 (Co-occurrence Matrices) 等方法提取底层视觉特征。对于中层语义特征，主要是利用行人的颜色、衣服、鞋子、头发长短和是否携带物品等语义信息。将这些语义信息加权与底层视觉信息结合，使得效果有所提升。对于高级视觉特征，主要的特征提取包括视频帧与帧之间的时序信息、图像光流特征、步态或移动轨迹等，如今已经提出了各种各样的方法。例如，Gou<sup>[10]</sup>等人提出 DynFV(Dynamic Fisher Vector) 特征和捕获步态和移动轨迹的 Fisher 向量编码的密集短轨迹时间金字塔特征，而 McLaughlin<sup>[11]</sup>等对图像提取颜色和光流特征，采用卷积神经网络处理得到高层表征，然后用循环神经网络 (Recurrent Neural Network, RNN) 捕捉时间信息，然后池化得到序列特征。

度量学习 (Metric Learning) 广泛应用于图像检索领域，因此在行人重识别中也得到了很多应用。度量学习根据同一个人的不同图片相似度大于不同行人的不同图片的原理，设计损失函数，使得相同行人图片的距离更小，而不同行人图片的距离更大。常用的损失函数包括对比损失 (Contrastive Loss)、三元组损失 (Triplet Loss)、四元组损失 (Quadruplet Loss)、难样本采样三元组损失 (triplet hard loss with batch hard mining, TriHard Loss)。不论是上述何种损失，其目标都是保证网络不仅能够在特征空间把正负样本推开，也能保证正样本对之间的距离更近。

局部特征是指对图像中的某一个区域进行特征提取，最后将多个局部特征融合起来作为最终特征。基于局部特征的 Re-ID 方法主要源于全局特征方法的瓶颈。常用的局部特征提取方法主要有图像切块、利用骨架关键点定位和姿态矫正等。

图像切块一般是指将人物 bounding box 图像垂直切割为若干份(如图 1.1 所示<sup>[12]</sup>), 分别输入网络进行训练。此外, 由于图像不对齐, 图像切片就会失效, 一些论文提出先利用预训练的人体姿态和骨架模型的先验知识, 估计出行人关键点后利用仿射变换将关键点对齐, 或者直接利用这些关键点来抠出感兴趣区域 (Region of Interest, ROI)<sup>[13]</sup>, 最终得到融合全局特征和局部特征的行人重识别特征, 从而提高效果。

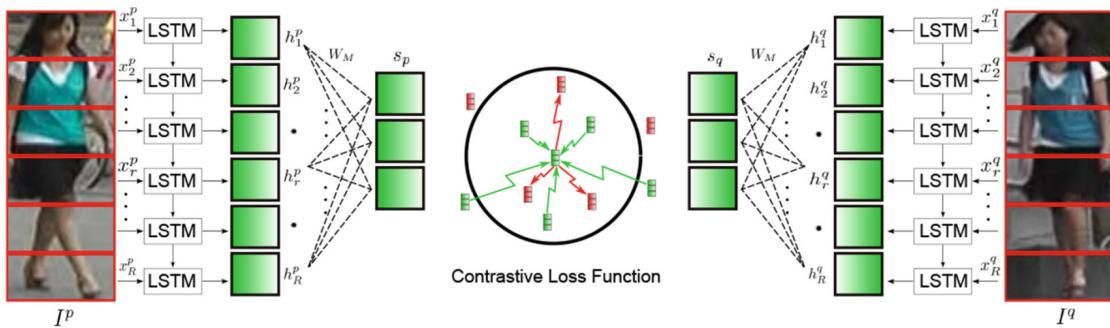


图 1.1 图片垂直切割后输入 LSTM(Long Short-Term Memory) 网络后进行特征提取

基于视频序列的 Re-ID 方法最主要特点是这类方法不仅考虑了图像的内容信息, 还考虑了帧与帧之间的运动信息、时序信息等。这类方法主要基于利用 CNN 提取空间特征的同时, 也要利用循环神经网络提取时序特征。累计运动背景网络 (Accumulative Motion Context Network, AMOC)<sup>[14]</sup>是代表方法之一, 其核心思想在于网络除了要提取序列图像的特征, 还要提取运动光流的运动特征, 最后再输入 RNN 来提取时序特征。于是, 通过 AMOC 网络, 每个图像序列都能被提取出一个融合了内容信息、运动信息的特征。

### 1.3.3 多摄像头行人位置估计算法

与单目行人检测算法相比, 多摄像头行人位置估计通常来看具有更好的效果, 因为多个视角很适合解决复杂遮挡问题<sup>[15]</sup>。

概率占据图方法 (Probabilistic Occupancy Map method, POM)<sup>[16]</sup>通过从多个视角中发掘其中的几何约束, 从而得到行人在地面的位置概率的生成模型。POM 是在平均场推断常常被自然地用于处理遮挡问题的背景下产生的。同时, 为了利用视频的时序信息, 概率占据图方法常常与凸最大成本流优化 (Convex Max-cost Flow Optimization)<sup>[17]</sup>相结合, 应用于行人跟踪的问题中。

在行人比较拥挤的场景中, Ge<sup>[18]</sup>等人提出采用随机人群配置的随机生成过程建模, 然后使用最大后验概率估计来寻找图像观测值的最佳拟合。而 Chavdarova<sup>[19]</sup>等人则提出了 DeepMCD(Deep Multi-camera Detection) 算法, 这种方法

集成了 CNN 得到的特征图，并展示了 CNN 分类器的准确性和可信度可以随着摄像头视角的增加而提高。为了缓解数据需求问题并提高泛化能力，作者首先使用较大的单目数据集——Caltech 数据集<sup>[20]</sup>来训练得到基础处理网络。之后，CNN 被用来使用从基础处理网络中得到的权重数据，并通过并行 (Multi-view Stream) 的方法处理得到最终的估计结果。作为这种方法的优化，Chavdarova 等人后续又在<sup>[19]</sup>中提出了两种多视角数据负挖掘的架构，来更好地训练这种网络。

目前，有一些多视角技术考虑对目标场景进行额外的训练步骤，以更好地利用应用场景中上下文信息。但由于对每个特定场景训练有着很大的隐藏成本，所以我们可以将其归类为不可泛化的方法<sup>[21]</sup>。在这个意义上，Baque 等人<sup>[15]</sup>提出将卷积神经网络 (CNN) 和条件随机场 (Conditional Random Field, CRF) 相结合的方法，以处理观察到的行人之间难以匹配的问题，即深度遮挡推理方法<sup>[15]</sup>。这种方法利用 CNN 和 CRF 相结合的体系结构和平均场推理，用类似<sup>[16]</sup>中的方式来生成概率占用图 (POM)，同时，还利用了 CNN 提取出来的判别特征。它引入了高阶 CRF 的方法，其中一元项由 CNN 的 ROI pooling 过程产生<sup>[22]</sup>，高阶项则是由解释遮挡的生成网络的预测和能判断身体部位图像块的 CNN 的预测的差异来计算。这个方法需要在特定的 WildTrack 数据集上训练的东西太多，其对每个地面计算检测框的精读会受到网格大小的限制；此外，WildTrack 数据集直接提供了地面分割的标注，但是一般的数据集只给位置不给地面分割的标注，导致这个方法在数据集上的应用有一定的局限性；同时，它检测框的估计需要先验知识，例如人的高度等。此外，MVDet<sup>[23]</sup>通过特征的透视变换将不同视角的行人热图放置在同一个坐标空间中，从而整合了多个视角的检测信息。类似的，DMCT<sup>[24]</sup>提出了透视感知网络 (Perspective-aware Network)，从而生成了与摄像机视角相关的探测区信息，之后，通过数据融合得到地面的占据热图 (Occupancy Heatmap) 估计，最后再利用 Deep Glimpse Network 得到行人的位置。

## 1.4 论文结构安排

本文的主要工作是完整搭建了一个多摄像头行人位置估计系统。这个系统包含单目行人检测、相机投影、对位置建立高斯概率分布模型、聚类进行数据融合、特征提取和训练等部分。最终在 WildTrack 数据集上达到了令人满意的行人位置估计结果。

第一步，行人检测系统。其功能是对输入的多个摄像头的 WildTrack 数据集

图片进行推断，以得到每张图中每个人行人的姿态节点，是后续计算行人位置的基础步骤。这一步选择了目前单目摄像头综合预测结果最好的 AlphaPose<sup>[25-27]</sup>模型。之后，提取 WildTrack 数据集中的各个摄像头视频帧，并进行去畸变和推断真值的预处理，作为训练集的扩充。最后利用 AlphaPose 对上述图片进行推断，得到行人姿态节点。

第二步，利用相机投影对位置建立高斯概率分布模型。在这一步中，先利用每个视角得到的人物姿态节点计算出每个视角每个人行人在当前图片中的 2D 足点(即双脚之间所站立的位置点)，再利用每个摄像头的相机参数以及相机投影关系，得到 3D 平面上每个行人的 3D 足点。最后，以每个 3D 足点为中心，结合摄像头视角方向，建立二维高斯概率分布作为对每个行人位置的估计。

第三步，利用层次聚类方法进行数据融合，并调整参数。将高斯概率之间的 KL 散度 (KL Divergence) 作为各个点之间的距离，利用层次聚类的方法，将每个行人在不同摄像头下得到的二维高斯概率分布进行融合，最终得到所有行人位置的粗略估计。通过对估计结果的优化，反过来调整前述二维高斯概率分布和层次聚类的超参数，从而优化后续步骤的输入。

第四步，生成热图和标签，然后进行训练和实验结果评测。首先，将第三步中得到的二维高斯概率分布处理为可供神经网络输入的热图，然后利用 WildTrack 数据集的实际标注得到与热图同样大小的黑白标签图。最后，输入分割模型 ResUnet++<sup>[28]</sup> 进行特征提取和训练，最终在 WildTrack 官方图片上测试，得到实验结果。

接下来对本文各个章节结构安排进行说明。

1. 第一章，引言。首先对本文所关注的问题进行描述，并对问题进行初步分解。然后阐述本课题研究的实际意义和价值，指出本文在这个问题中所关注的技术难点和突破口。最后对本课题当前的研究背景和现状加以说明，并对涉及到的相关技术和领域进行简要概括和梳理，主要包括单目行人检测算法、行人重识别算法和多摄像头行人位置估计算法。
2. 第二章，行人检测与高斯概率分布建模。本章中主要介绍主要工作中行人检测系统的应用、数据的预处理、数据集的扩充和利用相机投影对位置建立高斯概率分布模型的过程。首先将会介绍本文工作所采用的行人检测算法，以及该方法的优势，之后会介绍如何对 WildTrack 数据集进行扩充和预处理，包括真值推断和消除畸变等。最后，阐明利用相机参数进行相机投影和对位置建立高斯概率分布的原理和细节。

3. 第三章，数据准备与深度学习方法构建。本章中，首先介绍如何对各摄像头数据进行融合，并构建层次聚类方法得到粗略的行人位置估计，然后介绍如何通过粗略的估计结果反向优化高斯概率分布和层次聚类的超参数。同时，将会介绍利用扩充和预处理后得到的 WildTrack 数据集制作适合网络输入的热图和标签图。最后，将会对所采用的深度学习方法和神经网络进行说明。
4. 第四章，评价指标与实验结果。本章主要对结果的几个评价指标进行介绍，同时介绍实验结果，并对实验结果做出分析。
5. 第五章，结论。从本文所关注的问题和研究目的出发，对研究过程和结果进行总结，思考整个实验的不足之处，并提出未来可能的改进方向。

## 第 2 章 行人检测与高斯概率分布建模

### 2.1 数据集扩充和预处理

对于行人检测与高斯概率分布建模，首先需要选择合适的数据集，并对数据进行预处理。目前，比较适合多摄像头多目标位置估计的综合性能较好的数据集主要包括 Duke MTMC(Multi-Target, Multi-Camera) 数据集<sup>[29]</sup>和上文提到的 WildTrack 数据集。其中，Duke MTMC 数据集是 2014 年杜克大学校园拍摄的监控视频片段的数据集，用于视频跟踪系统、行人重识别和低分辨率面部识别的研发。这个数据集包含超过 14 小时的来自 8 个摄像头的同步监控视频，视频为 1080p 和 60fps，超过 200 万帧，其中共有 2000 名学生步行上下学。但由于英国《金融时报》调查发现此数据集可能侵犯个人隐私，Duke MTMC 数据集关闭了下载渠道。而 WildTrack 数据集则是一个记录了苏黎世联邦理工大学主楼外的学生的监控录像数据集。WildTrack 使用七个具有重叠视野的高科技静态定位摄像头（三个 GoPro Hero 4 和四个 GoPro Hero 3 摄像头）获取，具有高精度的联合摄像机校准以及视图序列同步，并且摄像头视野有着较大部分的重叠。因此可以满足在多视角多目标位置估计问题中大规模多摄像头数据集深度学习训练的需要。

本文采用 WildTrack 数据集。此数据集提供了时长为 35 分钟的分辨率为  $1920 \times 1080$ 、帧率为 60fps 的来自七个摄像头的同步视频，以及分辨率为  $1920 \times 1080$ 、帧率为 2fps 并消除畸变后的 400 张视频帧（如图 2.1 所示），并且提供了 400 张视频帧的行人位置标注，最后，还提供了每个摄像头的相机参数。此数据集所关注的区域为  $1200\text{cm} \times 3600\text{cm}$  广场，而在标注时，将广场分割为  $480 \times 1440$  的网格，每个网格边长为 2.5cm，从而得到 691200 个网格点。对每个网格点从上到下从左到右进行编号，可以得到网格编号，称为 Position ID。若在此网格上建立直角坐标系，坐标以米为单位，原点在  $(-3.0, -9.0)$  处，从而，根据 Position ID 可以得到标注坐标为

$$\begin{cases} X = -3.0 + 0.025 \times \text{Position ID \% } 480 \\ Y = -9.0 + 0.025 \times \text{Position ID / } 480 \end{cases}$$

由于后续神经网络训练的需要，而官方提供的 400 张图片和标注总体数量太少，难以达到预期训练效果，因此需要使用 WildTrack 数据集提供的视频进行数据集的扩充。由于带标注的 400 张图片帧率为 2fps，而视频序列的帧率则为 60fps，



图 2.1 WildTrack 数据集中来自七个摄像头的图片

因此，可以先从视频序列中找到与官方提供的图片集一致的帧，以此帧为起点，每 30 帧就会再次遇到官方图片集中的下一张图片。但是，由于从视频序列中提取的帧在 WildTrack 数据集中并不提供真值标注，因此通过对提供标注的 400 张图片集的标注进行插值，从而推断得到每两个图片之间提取到的视频帧的真值标注。此外，由于提供的图片集已经被去畸变处理过，而提取到的视频帧并未去畸变处理。故需要先对视频帧进行与图片集相同的去畸变处理，如图 2.2 所示。



图 2.2 去畸变处理示例

针孔摄像机会引起图像的严重畸变，主要有径向畸变和切向畸变。径向畸变使直线看起来是曲线。离图像中心越远，图像的径向畸变越大。径向畸变可以表

示为：

$$x_{\text{distorted}} = x(1 + k_1 r^2 + k_2 r^4 + k_3 r^6)$$

$$y_{\text{distorted}} = y(1 + k_1 r^2 + k_2 r^4 + k_3 r^6)$$

同样，切向畸变是由于摄像镜片与成像平面不完全平行造成的。因此，图像中的某些区域可能看起来比预期的更近。切向畸变量可表示为：

$$x_{\text{distorted}} = x + [2p_1 xy + p_2(r^2 + 2x^2)]$$

$$y_{\text{distorted}} = y + [p_1(r^2 + 2y^2) + 2p_2xy]$$

故需要得到五个畸变参数：

$$\text{distortion coefficients} = (k_1, k_2, p_1, p_2, k_3)$$

这些畸变参数一般需要标定求得。但在本文中，由于 WildTrack 数据集提供了每个相机的畸变参数，因此我们可以直接使用，无需标定。除此之外，还需要相机内外参数。其中，内部参数是特定于照相机的，包括像焦距 ( $f_x, f_y$ ) 和光学中心 ( $c_x, c_y$ ) 等信息，从而可以创建相机的内参数矩阵 (Camera Intrinsics)K，以消除由于特定相机的镜头畸变。相机矩阵是特定于相机的，因此得到后可以在同一相机拍摄的其他图像上重复使用。相机矩阵可表示如下：

$$K = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$$

而外部参数对应于旋转和平移矢量。实际上，WildTrack 数据集也已经提供了每个摄像机的相机内外参数。利用这些参数，我们可以通过经典的去畸变算法，实现对视频提取帧的去畸变操作。由于相机参数未变，所以去畸变处理后视频帧中背景状况与官方提供的图片集背景状况完全一致。至此，完成了数据的扩充和预处理操作。

## 2.2 行人检测算法

本文中的行人检测算法主要依赖于现成的不需要针对特定场景进行训练的单目行人检测器。从单目检测器得到的行人姿态节点中，我们可以计算出每个行人的足点（即双脚之间所站立的位置点）。后续便可以使用此足点进行数据融合，从而得到三维平面上的位置估计结果。

单目检测器实际上有很多算法，如引言中提到的基于运动检测的经典算法如

高斯混合模型分离算法、VIBE 算法和 CodeBook 算法等，但这些传统算法很容易受到环境和背景噪声的影响，而且不能检测静止目标，在机器学习方法迅速发展的今天已逐渐过时。因此，本文主要在机器学习算法尤其是深度学习方法中进行选取。在主要使用深度学习方法的算法中，一些单目检测器可以同时提供人物的检测框和姿态节点<sup>[30]</sup>。这些方法对全身姿态节点的使用可以更有效地处理遮挡问题，因此，我们选用了目前综合性能最好的 AlphaPose。在 AlphaPose 提供的多种模型中，本文选取了以 ResNet152 为骨干 (Backbone)，以 YOLOv3 为探测器，基于 MSCOCO 数据集<sup>[31]</sup>进行训练得到的 Fast Pose(DUC)<sup>[25-27]</sup>模型。这个模型将输入摄像头拍摄的尺寸为  $256 \times 192$  的图片，输出每张图片中每个人的姿态节点。其中，人物姿态节点的表示方式与 MSCOCO 数据集<sup>[31]</sup>中姿态表示相同，即包含 17 个姿态节点。在这些节点中，本文只使用了两个脚踝的节点，具体而言，保留了高于阈值  $t_s$  的脚踝节点。但由于脚踝节点高于地面，并不能代表人的位置，故可以利用两个脚踝节点计算得到一个抵消项  $\delta$ ，如下所示：

$$\delta = bb_{y_{max}} - \max(la_y, ra_y)$$

其中， $bb_{y_{max}}$  为当前行人的检测框下沿的纵坐标，而  $la_y$  和  $ra_y$  则分别是行人的左脚踝节点和右脚踝节点。因此，如图 2.3 所示，其中红色点为行人姿态节点，蓝色点为行人脚踝节点 (Ankle)，黄色点为脚踝中点 (Midpoint)，绿色点为足点 (GroundPoint)，可见，通过中点向下偏移  $\delta$  便可以计算得到行人的足点。根据上



图 2.3 行人足点计算实例

述方法，便可输入所有需要的图片或视频数据，推断出每帧中每个行人在此帧上的足点位置坐标，作为后续三维位置推断的基础。

## 2.3 相机投影原理

在行人检测中，得到每帧每个行人在当前帧的足点位置坐标后，接下来需要推断得到行人在实际地面上的坐标。这里主要使用几何约束和相机投影关系来实现。

相机成像的过程可表示为

$$\mathbf{p} = \mathbf{K} \mathbf{P}_c$$

其中， $\mathbf{p} = (\mu, \nu)$  是图像中像点的像素坐标， $\mathbf{K}$  为上文提到的相机内参数矩阵， $\mathbf{P}_c = (X_c, Y_c, Z_c)$  是相机坐标系下的三维点坐标。然而，相机坐标系并非“稳定”坐标系，因为这个坐标系会随着相机的移动而改变坐标的原点和各个坐标轴的方向。于是引进了稳定不变的世界坐标系。在世界坐标系中，设  $\mathbf{P}_w$  是在世界坐标系下的坐标，有

$$\mathbf{P}_c = \mathbf{R} \mathbf{P}_w + \mathbf{t}$$

其中， $\mathbf{R}$  为  $3 \times 3$  旋转矩阵， $\mathbf{t}$  为  $3 \times 1$  为平移向量。将上述公式写为齐次坐标的方式，如下所示：

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} = \begin{bmatrix} R_{11} & R_{12} & R_{13} & t_1 \\ R_{21} & R_{22} & R_{23} & t_2 \\ R_{31} & R_{32} & R_{33} & t_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix}$$

即

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^\top & 1 \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix}$$

于是推导得到相机的外参数 (Camera Exinsics)

$$T = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^\top & 1 \end{bmatrix}$$

将上式代入  $P_c = RP_w + t$  可得

$$\begin{bmatrix} \mu \\ \nu \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x & 0 \\ 0 & f_y & c_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R & t \\ 0^\top & 1 \end{bmatrix} \begin{bmatrix} X_W \\ Y_W \\ Z_W \\ 1 \end{bmatrix}$$

即得到了将真实场景中的三维点投影到二维的成像平面的相机投影关系。在上述关系中，我们可以令世界坐标系中的  $Z_w = 0$ ，从而得到从图像平面到实际地面的单应性矩阵 (Homography Matrix)。记相机外参数为  $T = [R|t]$ ，则图像平面上一点  $\mathbf{m} = (x, t)^\top$  到地面上一点  $\mathbf{M} = (X, Y, 0)^\top$  的映射关系由下式给出：

$$\begin{aligned} s \begin{pmatrix} \mu \\ \nu \\ 1 \end{pmatrix} &= K \begin{bmatrix} \mathbf{R}^1 & \mathbf{R}^2 & \mathbf{R}^3 & t \end{bmatrix} \begin{pmatrix} X \\ Y \\ 0 \\ 1 \end{pmatrix} \\ &= K \begin{bmatrix} \mathbf{R}^1 & \mathbf{R}^2 & t \end{bmatrix} \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix} \\ &= H^{-1} \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix} \end{aligned}$$

其中， $s$  为归一化参数， $\mathbf{R}^i$  是  $\mathbf{R}$  的列向量， $H$  为得到的单应性矩阵。依据上式，就可以将所得行人在图片上的足点投影到实际地面，得到在地面上的足点坐标。同时，需要舍弃那些投影在关注区域外的行人足点。另一方面，由于后续二维高斯概率分布建模的需要，此处还需要得出上文得到的每个行人地面足点所来源相机的视角方向。为了达到这一目标，我们可以令世界坐标系中的  $Z_w = 1$ ，则与上文方法相同，可以得到

$$\begin{aligned} s \begin{pmatrix} \mu \\ \nu \\ 1 \end{pmatrix} &= K \begin{bmatrix} \mathbf{R}^1 & \mathbf{R}^2 & \mathbf{R}^3 & t \end{bmatrix} \begin{pmatrix} X \\ Y \\ 1 \\ 1 \end{pmatrix} \\ &= [\mathbf{R}^1 \quad \mathbf{R}^2 \quad \mathbf{R}^3 + t]^{-1} K^{-1} \end{aligned}$$

于是，根据人物在图片上的足点可得到从摄像头到实际地面足点之间射线上的另一点，即射线上  $Z_w = 1$  的点。因此，容易得到此行人从来源相机到其地面足点射线的单位方向向量  $\mathbf{v}$ 。根据上文得到的行人足点和此方向向量便可以进行下一步中二维高斯概率分布的建模。

## 2.4 高斯概率分布建模

现在已经得到每个行人来源于每个相机在地面上的足点估计坐标。但很明显，上述方法比较比较简单，得到的位置估计准确性难以满足要求。另外，由于 WildTrack 数据集共有七个摄像头，因此每个行人在地面上的足点估计有七个，所以需要对地面上的足点进行融合，即把同一个人来自七个摄像头的足点坐标融合为一个坐标。在下一章中，我们将进行融合操作。由于当前人物在地面的位置估计并不准确，而其准确性主要与摄像机的距离和摄像机视角有关。可以想到，距离摄像机更近的足点置信水平更高，更远的点则置信水平会比较低。另一方面，由于单个摄像头视角固定，对一个行人左右水平位置的置信水平更高，而对其摄像头视角纵深方向位置的置信水平会比较低。综合来看，单个行人单个摄像头的足点估计可以由一个二维高斯概率分布 (Two-dimensional Gaussian distribution) 来估计，此二维高斯概率分布的形状大致呈长轴方向为摄像头视角纵深方向、短轴方向为摄像头横向的椭圆形。

二维高斯概率分布的密度函数为：

$$f(x, y) = \frac{1}{2\pi\sigma_1\sigma_2\sqrt{1-\rho^2}} e^{-\frac{1}{2(1-\rho^2)}\left(\frac{(x-\mu_1)^2}{\sigma_1^2} - \frac{2\rho(x-\mu_1)(y-\mu_2)}{\sigma_1\sigma_2} + \frac{(y-\mu_2)^2}{\sigma_2^2}\right)}$$

其中  $\mu_1, \mu_2, \sigma_1, \sigma_2, \rho$  都是常数，可记作：

$$\boldsymbol{\mu} = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}, \Sigma = \begin{pmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{pmatrix}$$

可通过上述两个参数确定此二维高斯概率分布。对于当前来源于特定相机的特定行人足点进行建模，根据相机投影关系，容易得到高斯概率中心点  $\boldsymbol{\mu}$  即为其地面足点坐标。由于当前行人足点从来源相机到其地面足点射线的单位方向向量为  $\mathbf{v}$ ，记其垂直方向的单位方向向量为  $\mathbf{v}'$ ，可令

$$C = k_c \mathbf{v}^\top \mathbf{v}, C' = k_c \mathbf{v}'^\top \mathbf{v}'$$

则二维高斯概率分布中

$$\begin{cases} \mu = (X_w, Y_w)^\top \\ \Sigma = kC + (1 - k)C' \end{cases}$$

其中  $k$  为高斯概率形状调整参数，若  $k \rightarrow 1$ ，则二维高斯概率趋于扁平，长轴方向为  $\boldsymbol{v}$ ，若  $k \rightarrow 0$ ，则二维高斯概率趋于扁平，长轴方向为  $\boldsymbol{v}'$ ，若  $k \rightarrow 0.5$ ，则二维高斯概率趋于一个圆形。从而，根据上述方法可以建立单个行人单个摄像头的足点估计的二维高斯概率分布，对于单个摄像头的建模结果如图 2.4(a)所示，其中黄色三角为真实标注点，不同颜色的分布为高斯概率分布。对于 WildTrack 数据集，将七个摄像头所得行人在地面上的足点进行二维高斯概率分布建模后，投影在同一张图上，如图 2.4(b)所示，其中黄色三角为真实标注点，相同颜色的椭圆为同一摄像头来源的二维高斯概率分布建模。

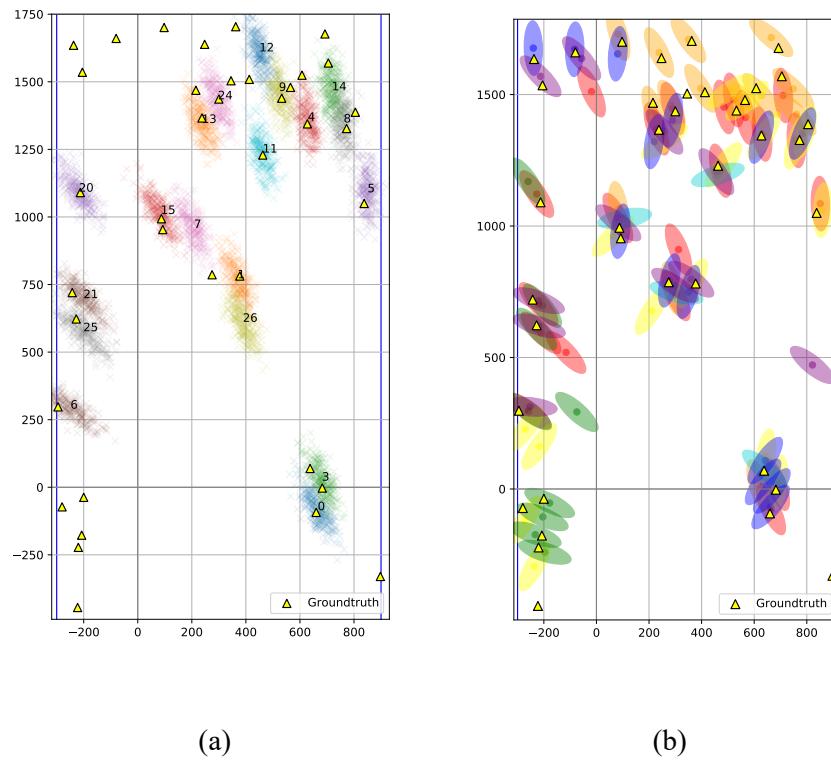


图 2.4 多个分图的示例

## 2.5 本章小结

本章主要内容为深度学习算法的前置步骤。首先介绍了数据集的扩充和预处理过程，然后说明了所使用的现成行人检测算法的选取和具体实现细节，之后比较详细地介绍了相机投影原理和如何利用相机投影对得到的行人姿态节点进行处理，从而计算出行人的实际地面位置坐标，最后，详细介绍了所采用的二维高斯概率分布建模原理，阐明了所得行人地面位置坐标和相机参数如何决定了高斯概率分布的参数。这些步骤，为后续的数据融合和深度学习算法的使用做了铺垫。

## 第3章 数据准备与深度学习方法构建

上一章中，我们对行人在实际地面的位置进行了二维高斯概率分布建模，而在这一章中，我们将利用层次聚类的方法融合不同摄像头的数据，并利用深度学习网络对 WildTrack 数据集场景进行学习，最终得到更精确的行人位置估计结果。

### 3.1 层次聚类方法

在上文中的二维高斯概率分布建模中，对于每个行人每个摄像头的数据均进行了一次建模过程。因此，对于每个行人的位置，有来自多个摄像头的二维高斯概率分布来建模其在特定摄像机视角下的位置概率分布。于是，我们需要对每个行人将其来自多个摄像头的位置概率进行融合，最终可得到一个确定的预测点。以此点与真实地面坐标的差异为指标，可以反向优化前述二维高斯概率分布建模时和数据融合时的超参数。令预测点与真实地面坐标的差异趋于最小值，则可得到最优的超参数。利用此最优超参数生成后续神经网络输入的热图，可以最大化神经网络方法的效果。

本文中对于相同行人来自不同摄像头的二维高斯概率分布建模融合方式为层次聚类的方法，其优势主要是模型的解释能力比较强。层次聚类包含两种方式，即凝聚型层次聚类和分裂型层次聚类。本文中使用自下而上的凝聚型算法，其核心原理是：将每个二维高斯概率分布作为初始节点，初始假设每个节点都是一类，每一次迭代将会聚合最相近的两个点或聚类，当所有点或聚类都合并成一类或者满足停止条件时，则终止模型迭代，此时便得到了聚类结果。

在聚类过程中，我们采用了 Alejandro López-Cifuentes<sup>[32]</sup>等人提出的在融合地面点时的两个约束条件。即，属于同一个行人的所有二维高斯概率分布都需满足：

1. 来自不同的摄像机 (因为每帧中一个行人只可能出现一次);
2. 二者之间距离必须低于阈值  $t_g$ 。

同时，在层次聚类时，计算两个组合数据点间距离的方法有三种，分别为 Single-Linkage, Complete-Linkage 和 Average-Linkage。在选择合适的距离计算方法之前，我们先来介绍下这三种计算方法以及各自的优缺点。

- Single-Linkage：将两个不同聚类的数据点中距离最近的两个数据点间的距

离作为这两个聚类之间的距离。这种方法有一个缺陷，即容易受到极端值的影响。两个距离并不近的聚类，可能会因为其中两个极端点距离过近而融合，影响最终聚合效果；

- **Average-Linkage:** 将两个不同聚类的数据点中距离最远的两个数据点间的距离作为这两个组合数据点的距离。这种方法的缺陷与 Single-Linkage 相反，即两个距离比较近的聚类可能由于其中的极端值距离较远而无法聚类；
- **Complete-Linkage:** 这种距离的计算方法是计算两个聚类的每个数据点与其他所有数据点的距离，将所有距离均值作为两个聚类间距离。这种方法计算量比较大，但结果比前两种方法更合理。

因此，本文层次聚类中选择了更合理的 Complete-Linkage。同时，二维高斯概率分布之间的距离若直接用中心点之间的距离来衡量，则会损失其所估计的位置概率信息，于是本文采用相对熵 (Relative Entropy)，又被称为 KL 散度 (Kullback-Leibler Divergence)，来度量两个概率分布 (Probability Distribution) 间差异。在信息理论中，相对熵等价于两个概率分布的信息熵 (Shannon Entropy) 的差值。设  $P(x), Q(x)$  是随机变量  $X$  上的两个概率分布，在连续随机变量的情形下，相对熵的定义为：

$$KL(P\|Q) = \int P(x) \log \frac{P(x)}{Q(x)} dx$$

但由于此处所有概率分布均为二维高斯概率分布，故它们之间的 KL 散度可以方便地用下式来计算：

$$\begin{aligned} & KL(N(x|\boldsymbol{\mu}_1, \Sigma_1) | N(x|\boldsymbol{\mu}_2, \Sigma_2)) \\ &= \frac{1}{2} \left[ \log \frac{|\Sigma_2|}{|\Sigma_1|} - K + \text{tr}(\Sigma_2^{-1} \Sigma_1) + (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^\top \Sigma_2^{-1} (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2) \right] \end{aligned}$$

其中， $N(x|\boldsymbol{\mu}_1, \Sigma_1)$  为  $\boldsymbol{\mu} = \boldsymbol{\mu}_1, \Sigma = \Sigma_1$  的二维高斯概率分布， $N(x|\boldsymbol{\mu}_2, \Sigma_2)$  为  $\boldsymbol{\mu} = \boldsymbol{\mu}_2, \Sigma = \Sigma_2$  的二维高斯概率分布。最后，得到层次聚类结果后，由于相机有重叠的视野，所以我们丢弃只有一个数据点的聚类，从而减少了错误聚类的数量，提升了聚类效果。最终得到的层次聚类结果如图 3.1(a)所示，其中黄色三角形为行人实际地面位置，而橙色星形为层次聚类得到的行人地面位置估计。作为补充，图 3.1(b)中也显示了作为层次聚类输入的人物位置二维高斯概率分布 (以椭圆形表示)，其中来自同一摄像头的分布用同一种颜色表示，同一颜色、同一序号的二维高斯概率分布为同一聚类。

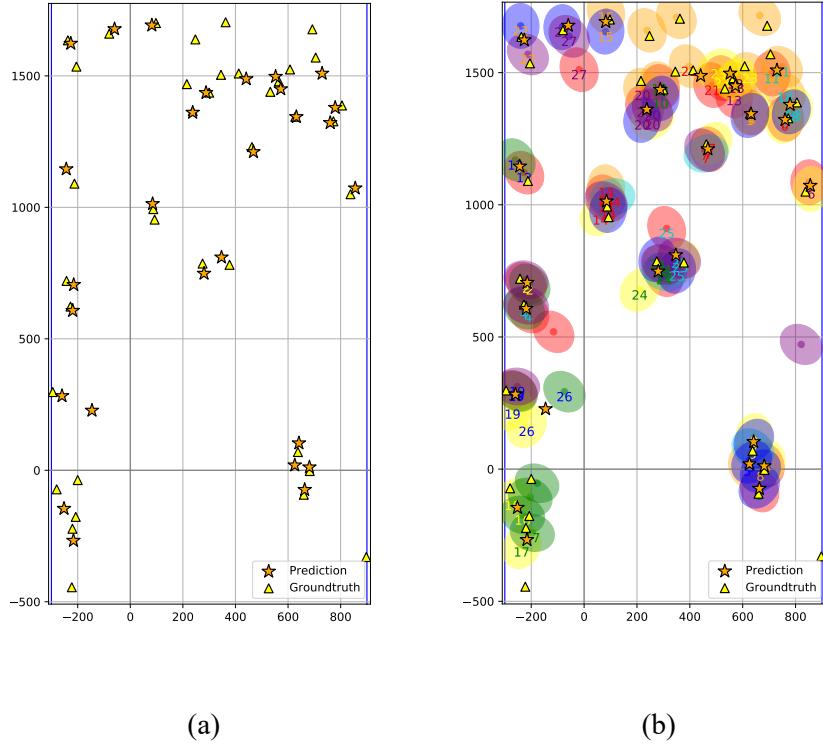


图 3.1 层次聚类结果

### 3.2 深度学习数据准备

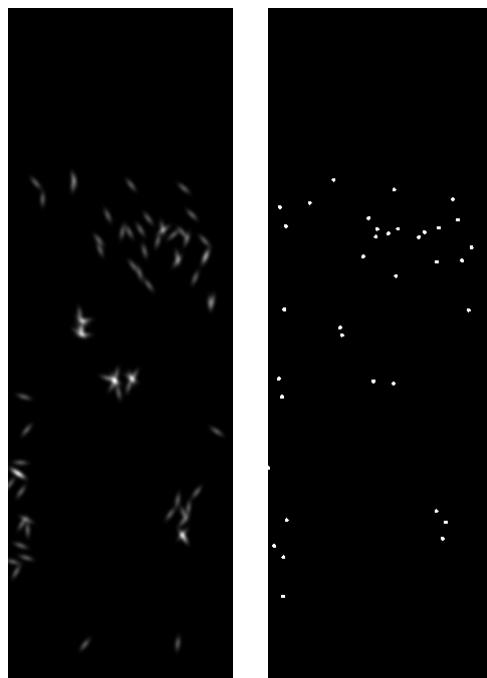
对二维高斯概率分布进行聚类融合之后，接下来需要生成适用于后续神经网络输入和输出的图片。对于输入，本文将前述得到的二维高斯概率分布投影到一张大小为  $224 \times 672$  的热图 (Heatmap) 中，便于神经网络处理。为了得到最优的热图，需要对前述生成二维高斯概率分布的超参数进行调整。调整的方法是，令层次聚类得到的聚类中心点与真实地面坐标的差异趋于最小值，在此过程中进行参数记录，则可得到最优的超参数如下表 3.1 所示。利用此最优超参数生成后续神经网络输入的热图，可以最大化神经网络方法的效果。

接下来，使用上述参数得到二维高斯概率分布后，将其投影到热图中。而输入神经网络中的真值标签图，则可以使用真实地面坐标的占据图 (Occupancy Map)<sup>[16]</sup> 表示。最终所得热图如图 3.2(a) 所示，标签图如 3.2(b) 所示。

最后，在输入神经网络前，还需要划分数据集。本文将上文中数据集扩充时通过提取的视频帧作为训练集，将官方提供的 400 帧图片集作为测试集。

表 3.1 反向优化得到的最优超参数

参数	值	描述
$k_c$	700	二维高斯概率分布中 $C = k_c \mathbf{v}^\top \mathbf{v}, C' = k_c \mathbf{v}'^\top \mathbf{v}'$ 中的 $k_c$
$k$	0.85	二维高斯概率分布中 $\Sigma = kC + (1 - k)C'$ 中的 $k$
$t_g$	19cm	层次聚类中聚为一类的阈值距离
$t_{ankle}$	0.35	单目探测器结果中保留概率超过 $t_{ankle}$ 的脚踝节点



(a) 热图 (b) 标签图

图 3.2 神经网络输入与标签

### 3.3 深度学习神经网络

#### 3.3.1 神经网络选择

现在，我们的目标是利用深度学习算法对输入的位置概率特征进行学习，最终得到能够准确得到行人的位置。具体来说，即输入高斯概率分布热图，输出对应行人位置的占据图 (Occupancy Map)。目前，在图像分析领域，深度学习方法实现的语义分割取得了很大的进展<sup>[33-35]</sup>，本文也拓展了语义分割在行人位置估计上的应用。在生物医学应用的语义分割领域，一种流行的深度学习体系结构是 U-Net<sup>[34]</sup>，它在 2015 年 ISBI 细胞跟踪挑战上达到了最好的性能，而 ResUNet(Residual U-Net)<sup>[35]</sup> 结构作为 U-Net 结构的改进，在道路图像提取中达到了最优的性能。同时，ResUNet++<sup>[36]</sup> 以 ResUNet 为基础发展而来，在医学图像分割领域取得了很不错的成绩。本文基于以上进展，将上述网络作为本文工作的基础，实现了 ResUNet 和 ResUNet++ 两种架构，将其作为多摄像头多目标行人位置估计的深度学习方法部分。

#### 3.3.2 神经网络结构

本文所实现的 ResUNet 使用了 U-Net、残差块等结构 (Residual Units)，而 ResUNet++ 则在 ResUNet 的基础上加入了挤压和激励单元 (Squeeze and Excitation Units)、空洞空间金字塔池化 (Atrous Spatial Pyramid Pooling, ASPP) 和注意力机制 (Attention Units)，从而提升了效果。下文对这些结构一一介绍，并阐明其在网络中的作用。

**U-Net:** 在语义分割中，为了获得更好效果，在保留高级语义信息的同时必须使用低级信息细节。然而，训练这样一个深度的神经网络是很困难的，尤其是当训练样本很有限时。解决此问题的一种方法是 Long 等人<sup>[37]</sup> 提出的先使用预训练的网络，然后在目标数据集上对其进行微调。而另一种方法则是采用 U-Net<sup>[34]</sup> 中使用的数据扩充方法。此外，U-Net 的体系结构也有助于缓解训练的困难，这是因为将低级特征复制到相应的高级特征上，实际上创建了一条信息传播路径，允许信号更容易地在低级和高级特征之间传播。这样不仅有助于训练的反向传播过程，而且还可以将低级特征中的细节补偿到高级语义特征上去。这种特点和 ResNet<sup>[38]</sup> 中提出的 Residual Units 原理很相似。可见，U-Net 结构是一种适合本文中最后部分位置估计的网络基础结构。

**残差块 (Residual Units):** 在 ResNet 网络提出之前，传统的卷积神经网络都是通过将一系列卷积层与池化层进行堆叠得到的。一般认为，网络越深，特征信息

越丰富，模型效果应该越好。但是实验证明，效果并非如此。这是因为当网络深度堆叠到一定程度时，会发生梯度消失或梯度爆炸的问题和退化问题 (Degradation Problem)<sup>[38]</sup>。因此，何凯明等人提出了残差块来解决此问题。残差块的功能可由下式表示：

$$\begin{aligned}\mathbf{y}_l &= h(\mathbf{x}_l) + \mathcal{F}(\mathbf{x}_l, \mathcal{W}_l), \\ \mathbf{x}_{l+1} &= f(\mathbf{y}_l),\end{aligned}$$

其中， $\mathbf{x}_l$  和  $\mathbf{x}_{l+1}$  是第  $l$  个残差单元的输入和预测， $\mathcal{F}(\cdot)$  为残差函数， $f(\mathbf{y}_l)$  是激活函数， $h(\mathbf{x}_l)$  是恒等映射函数，例如  $h(\mathbf{x}_l) = \mathbf{x}_l$  这样的函数就是恒等映射函数。在一个残差块中，有以不同方式组合的批标准化 (Batch Normalization)，ReLU 激活函数和卷积层，而本文中采用了何凯明<sup>[38]</sup>提出的综合性能最好的组合方式，图 3.3 展示了未采用残差块与采用残差块的对比，同时展示了所采用残差块的结构。

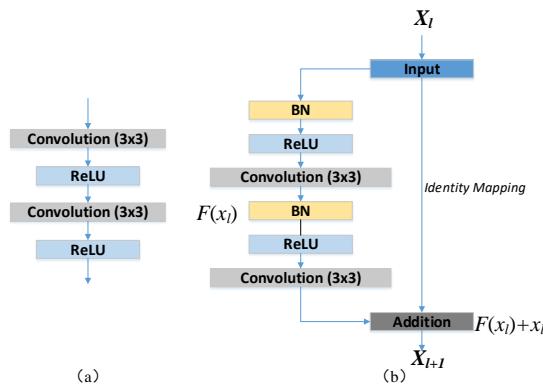


图 3.3 (a) 未采用残差块与 (b) 采用残差块的对比

**ResUNet:** 结合上述 U-Net 和残差块的技术，可以构建 ResUNet 网络<sup>[35]</sup>。这种结合可以带来两个优势：1) 残差块使得可以构建的网络深度大大增加；2) 残差块内部和网络高、底层次之间的直接连接大大减少了信息传播中的退化现象，从而在提高效果的同时还可以减少网络参数。本文所实现的 ResUNet 网络结构如图 3.4(a)所示。网络采用了七层结构，为自编码器结构，主要分为编码层、中间层和解码层。编码层将输入的图片编码为压缩抽象表示，而解码层将压缩抽象表示解码为与输入大小相同的输出图片，中间层主要起到连接编码层和解码层、传输特征的作用。

**挤压和激励单元 (Squeeze and Excitation Units):** 挤压和激励网络 (Squeeze-and-Excitation Networks)<sup>[39]</sup>通过精确建模网络中通道的相互依赖关系，以及校准特征的方法，大大提高了网络的表示性能。因此，本文所采用的 ResUNet++ 中也

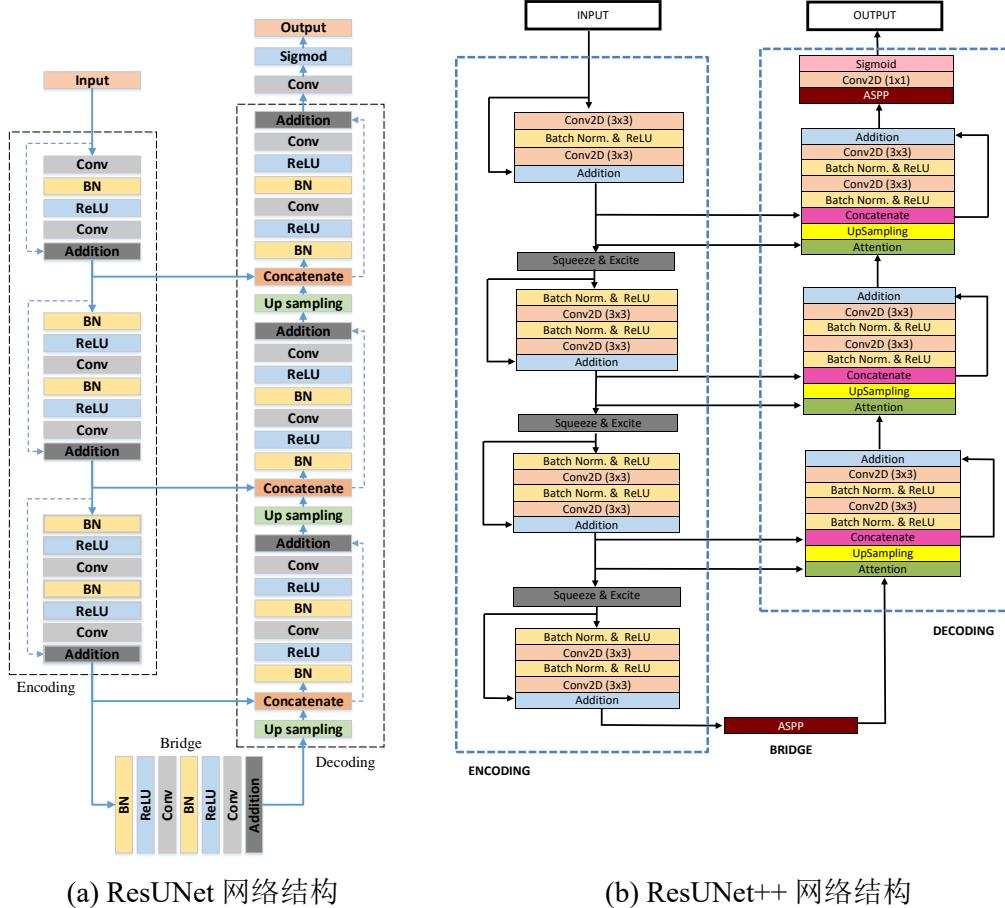


图 3.4 本文实现的两种神经网络结构

使用了挤压和激励单元，其主要目的是提高网络对关注特征的敏感性，抑制不必要的特征。正如其名，挤压和激励单元实施的操作为先挤压全局信息并通过全局平均池 (Global Average Pooling) 来生成信息，然后激励以得到通道之间的相关性。

**空洞空间金字塔池化 (Atrous Spatial Pyramid Pooling, ASPP):** 何凯明等人<sup>[40]</sup>提出的 ASPP 方法，使得网络可以控制视野从而精确捕获多个尺度的信息。在本文实现的 ResUNet++ 中，ASPP 主要作为编码器和解码器之间的中间层，其作用主要是为语义分割提取多尺度的信息。

**注意力机制 (Attention Units):** 注意力机制实质上是一种分配机制，其核心思想是突出对象的某些重要特征，使得网络能够根据对象的重要程度重新分配资源。注意力机制的主要优势是可以被简单地运用在任何尺寸的输入中，显著提升特征的质量，从而促进效果的提升。在本文实现的 ResUNet++ 中，注意力机制被加入到解码器的部分，从而使得网络能够专注于特征图中的关键区域。

**ResUNet++:** 结合上述挤压和激励单元、空洞空间金字塔池化和注意力机制，

在 ResUNet 的基础上，本文实现了 ResUNet++ 网络。网络也分为七层结构，与 ResUNet 的结构基本一致，但加入了挤压和激励单元、空洞空间金字塔池化和注意力机制，从而大大提升了网络的预测效果。

### 3.3.3 损失函数

在构建的 ResUNet 和 ResUNet++ 两个深度学习神经网络中，本文均采用了二进制交叉熵 (Binary Cross Entropy, BCE) 损失和 Dice 系数 (Dice Coefficient) 损失相结合的损失函数。二进制交叉熵可由下式计算：

$$\mathcal{L}_{BCE} = \frac{1}{N} \sum_{n=1}^N l_n$$

其中， $l_n = -w[y_n \cdot \log x_n + (1 - y_n) \cdot \log(1 - x_n)]$  为第  $n$  个样本对应的 loss，其中  $w$  为超参数。在本文中，上式 loss 体现为将预测图的第  $n$  个像素作为样本，标签图的第  $n$  个像素作为样本标签的 loss。而 Dice 系数是一种集合相似度度量函数，通常用于计算两个样本的相似度。Dice 系数损失的计算可由下式得到：

$$\mathcal{L}_{Dice} = 1 - \frac{2|X \cap Y|}{|X| + |Y|}$$

其中， $X$  为预测图， $Y$  为标签图， $|X \cap Y|$  是  $X$  和  $Y$  之间的交集， $|X|$  和  $|Y|$  表示  $X$  和  $Y$  的元素的个数。在实际计算时， $|X \cap Y|$  近似为预测图和标签图之间的点乘，并将点乘的结果逐元素相加。而计算  $|X|$  和  $|Y|$  时，本文采取了直接进行逐元素相加的方式。

最终，将两种损失结合起来，就得到了本文所采用的损失函数，如下：

$$\mathcal{L} = \mathcal{L}_{BCE} + \mathcal{L}_{Dice}$$

## 3.4 本章小结

本章中介绍了构建层次聚类方法得到粗略的行人位置估计和通过粗略的估计结果反向优化高斯概率分布和层次聚类的超参数的过程，然后介绍了利用扩充和预处理后得到的数据如何被用来制作适合网络输入的热图和标签图。最后，详细阐述了本文所实现的 ResUNet 和 ResUNet++ 两种深度学习神经网络结构和所采用的损失函数。这两种深度学习方法将输出行人位置估计的更准确的预测，可以作为本文工作中的估计结果。

## 第4章 评价指标与实验结果

本章首先介绍本文所实现的多摄像头多目标行人位置估计算法中所采用的实验评价指标，然后详细阐述实验过程和实验的环境、参数。最后，将实验结果与目前业界最新算法进行比较，并提出未来可能的改进建议。

### 4.1 评价指标

精确率 (Precision) 和召回率 (Recall) 是广泛用于信息检索和统计学分类领域的两个度量值，用来评价结果的质量。因此，本文采用了精确率、召回率来衡量最终位置估计效果。然而，由于在大规模数据集合中，精确率和召回率两个指标往往是相互制约的，因此，本文还采用了 F 分数 (F-score) 来综合权衡精确率和召回率。即，本文共使用了精确率、召回率和 F 分数三个评价指标来衡量最终的位置估计效果。

为了得到上述三个评价指标，首先需要得到实验结果的混淆矩阵 (Confusion Matrix)。混淆矩阵也称误差矩阵，是表示精度评价的一种标准格式，在本文中为两行两列的矩阵形式。其主要参数有四个，即真阳性 (True Positives, TP)、真阴性 (True Negatives, TN)、假阳性 (False Positives, FP)、假阴性 (False Negatives, FN)。在本文实验中，TP 集合为与实际位置相符的预测点；TN 集合应为预测点之外的点中排除实际位置的点，由于计算上述三个评价指标并没有用到 TN 集合，故被忽略；FP 集合为预测点中没有命中实际位置的点；FN 集合为没有被预测到的实际位置点。在本实验中，预测点与实际位置命中的判定半径为  $r = 0.5m$ ，即若预测点与实际点距离相聚小于 50cm 即判定为成功估计了行人的位置。

上文提到，计算混淆矩阵的参数时，需要判定预测点与实际位置点是否对应，从而判定是否正确预测。现在，我们得到了对行人位置在地面上的最终预测点，同时，我们有行人地面实际位置的标注，那么就需要将这些预测点和实际标注一一匹配起来，从而可以计算预测点与实际点之间的距离，以判断是否预测成功。解决上述匹配问题的方法可以抽象为图论中寻找最大匹配的算法，在本文中，上述问题抽象为在二分图中寻找最大匹配。因此，我们实现了匈牙利算法 (Hungarian Algorithm) 来解决这一问题。匈牙利算法利用深度优先搜索来不断寻找增广路。它从每个未匹配的实际地面点开始，如果直接找到了预测点中未匹配的点，则直

接返回此匹配路径；若找到预测点已匹配的点，那么从该点匹配的点出发继续深度优先搜索未匹配点。

得到混淆矩阵后，可以计算上述三个评价指标。

- 精确率 (Precision) 精确率是针对预测结果而言的，它表示的是预测点中有多少是预测正确的。精确率显示了模型预测点的精确性，可由下式计算：

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

- 召回率 (Recall) 召回率是针对原来的样本而言的，它表示的是实际位置点中有多少被预测命中了。召回率显示了模型是否能够估计出视野中更多行人的实际位置，其可由下式计算：

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

- F 分数 (F-score) 精确率和召回率指标有时候会出现矛盾的情况，这样就需要综合考虑二者，最常见的方法就是利用 F 分数，即精确率和召回率的加权调和平均：

$$\text{F-score} = (1 + \beta^2) \cdot \frac{\text{Precision} \cdot \text{Recall}}{\beta^2 \cdot \text{Precision} + \text{Recall}}$$

本文中令  $\beta = 1$ ，即实际上使用 F1-score。

## 4.2 实验设置和结果分析

我们利用了表 3.1 中的超参数进行层次聚类的数据融合，之后可以初步得到行人位置估计结果，这些实验结果与深度学习方法的比较如表 4.1 所示。

在深度学习方法中，主要在 WildTrack 数据集上进行实验。其中，作为训练集的是从 WildTrack 数据集提供的七个摄像头视频提取到的视频帧，每个摄像头共有 11558 张图片，其中真值由官方提供的 400 张图片真值插值得到；作为测试集的是 WildTrack 数据集提供的每个摄像头 400 张的去畸变图，提供了真值标注。实现的 ResUNet 和 ResUNet++ 网络均基于 PyTorch，部署在 NVIDIA RTX 3090 GPU 进行训练，系统版本为 Ubuntu 18.04，提供 CUDA 11.3 支持。在训练的过程中，输入热图和输出预测图的尺寸均为  $224 \times 672$ ，输入网络时采用的 Batch Size 为 4，使用 Adam 优化器进行调优。同时，为了避免网络难以收敛到最优状态，初始学习率设置为  $1e-3$ ，每 20 个训练周期 (epoch) 学习率衰减为原来的 0.1 倍，一共训练 120 个周期。

最后，将深度学习方法中 ResUNet 和 ResUNet++ 网络的实验结果与目前领域内几个深度学习方法的比较如下表 4.1 所示。

表 4.1 层次聚类方法、ResUNet 和 ResUNet++ 与其他方法实验效果比较

方法	精确率	召回率	F 分数
Deep-Occlusion <sup>[15]</sup>	95%	80%	86%
Generalized Mutiview Detection <sup>[41]</sup>	93%	94%	93%
Hierarchical Clustering	62%	87%	72%
ResUNet	70%	92%	79%
ResUNet++	82%	98%	89%

由表 4.1 可以看出，ResUNet++ 网络结构效果比 ResUNet 效果好很多，而没有使用神经网络的层次聚类融合方法效果最差。同时，也可以发现，ResUNet++ 在多摄像头多目标行人位置估计领域，在未使用行人重识别和时序信息的情况下，达到了接近目前业界领先算法的效果。

### 4.3 本章小结

本章中主要介绍了实验的评价指标、实验设置细节和具体实验结果的分析。实验的评价指标主要使用了精确率、召回率和 F 分数三个指标来衡量算法位置估计的效果，实验设置上，对数据集进行扩充以增加训练量，同时利用学习率衰减的训练策略，提升了训练效果。在实验结果方面，仅仅使用层次聚类来得到行人的位置估计的算法效果并不好，而在层次聚类的基础上利用 ResUNet 和 ResUNet++ 的深度学习网络可以达到较好的效果。此外，所实现的 ResUNet++ 在 ResUNet 基础上改进而来，达到了接近目前最优算法的性能。

## 第 5 章 结论与展望

### 5.1 结论

由于社会经济和科技的迅速发展，近年来摄像头在生活中已经越来越普遍。多摄像头行人位置估计是计算机视觉领域一个重要发展方向，今年来在人工智能和深度学习方法迅速发展的背景下取得了长足进步。在当今摄像头无处不在的背景下，多摄像头行人位置估计的应用场景也越来越多，社会对这项技术的需求也越来越大。这些行人位置估计算法的发展和成熟使得我们可以高效准确地在很多公共场所对行人进行跟踪或轨迹的预测，可以促进自动驾驶技术的进一步发展，甚至还可以应用于电影和游戏行业，促进虚拟现实技术的发展。可见，多摄像头行人位置估计算法很有研究的意义和价值。

本文所提出的多摄像头行人位置估计算法可以分为四个部分：行人检测、位置的相机投影与建模、数据关联和融合、特征学习与输出结果。其中行人检测直接使用即插即用的 Alphapose 中的预训练网络提取人体姿态节点，从而可以结合外观信息迅速得到行人的足点。此后，利用相机投影的原理将行人足点转为在实际地面上的二维高斯概率分布，从而巧妙地结合了不同视角的摄像机中不同行人位置不同的概率分布信息。在数据关联与融合阶段，利用 KL 散度作为距离，将地面上的行人二维高斯概率分布利用层次聚类的方式进行融合，再利用融合得到的粗略位置估计结果反向调整前面步骤的参数。这样，将不同视角摄像机的位置概率分布信息进行了很好的融合，有效地降低了遮挡带来的估计效果减退。最后，实现的 ResUNet 利用了 U-Net 和残差块的优势，在加深网络深度的同时减少了退化的现象。而在 ResUNet 基础上实现的 ResUNet++ 则加入了挤压和激励单元、空洞空间金字塔池化和注意力机制，提高了网络的表示性能，同时使得网络能够控制视野、在更重要的特征上投入更多的关注度。ResUNet++ 在得到的二维高斯概率分布信息上训练，学习到了不同视角摄像机中的行人位置概率分布特征，提升了模型预测效果，接近业界先进水平。

### 5.2 展望

本文算法可以接近业界先进水平，但由于时间有限，未能继续在此算法中继续拓展和完善来推动其达到领先水平。在将来的工作中，我们还可以在以下方面

作新的尝试，从而改进算法、提升效果。首先，目前直接使用的 Alphapose 需要将高清摄像头输入压缩为  $256 \times 192$  尺寸，因此在这个过程中损失了很多原有信息，未来可以先识别出人物得到其检测框后，将每个行人从高清图中截取下来，分别输入到姿态检测的神经网络中，这样便可以保留更多图像信息，提高每个人足点的预测精度。此外，当前算法，针对同一摄像机而言，行人与摄像机的距离远近对其二维高斯概率分布建模几乎没有影响，但众所周知，近处行人的位置概率分布区域显然应该比远处行人的概率分布更小，近处的行人应该有更精确的位置估计。因此，未来可以加入这一影响参数，也可以加入一个权重系数，使得距离摄像头更近的节点有更高的权重，从而提高位置估计精度。此外，本文算法对行人的外表特征并没有显式地利用，对于视频序列中的时序信息也未加以利用，未来可以加入行人重识别算法的部分功能，同时利用多个连续帧中同一行人位置变化连续的特点，进一步提升神经网络的预测效果。最后，还可以在网络的训练过程中利用数据集增强的方法，如裁剪、加入噪声等方法，进一步扩充数据集，以提高网络的适应性，另外，还可以加入权重衰减 (Weight Decay) 方法，尝试更多的训练策略，以使得网络发挥其最大性能。可见，此课题仍有很大的发展和探索空间。最后，多摄像头行人位置估计还有很多可以探索的方向，例如，目前大多数多摄像头行人位置估计算法只能在特定数据集上训练和应用，但是实际生活中的场景是非常多样的，因此探索一个可拓展的，即在一个场景训练便可以应用于大多数场景的深度学习网络是很有必要的，此外，当前算法性能相对较差，难以达到实时预测的效果，未来可以从提高算法性能的角度出发，探索更迅速的位置估计算法。

## 参考文献

- [1] Chavdarova T, Baqué P, Bouquet S, et al. Wildtrack: A multi-camera hd dataset for dense unscripted pedestrian detection[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 5030-5039.
- [2] Zivkovic Z, Van Der Heijden F. Efficient adaptive density estimation per image pixel for the task of background subtraction[J]. Pattern recognition letters, 2006, 27(7): 773-780.
- [3] Barnich O, Van Droogenbroeck M. Vibe: A universal background subtraction algorithm for video sequences[J]. IEEE Transactions on Image processing, 2010, 20(6): 1709-1724.
- [4] Kim K, Chalidabhongse T H, Harwood D, et al. Real-time foreground–background segmentation using codebook model[J]. Real-time imaging, 2005, 11(3): 172-185.
- [5] Dalal N, Triggs B. Histograms of oriented gradients for human detection[C]//2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05): volume 1. Ieee, 2005: 886-893.
- [6] Dollár P, Tu Z, Perona P, et al. Integral channel features[M]. BMVC Press, 2009.
- [7] Chen Y, Tai L, Sun K, et al. Monopair: Monocular 3d object detection using pairwise spatial relationships[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 12093-12102.
- [8] Bertoni L, Kreiss S, Alahi A. Monoloco: Monocular 3d pedestrian localization and uncertainty estimation[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019: 6861-6871.
- [9] Hayakawa J, Dariush B. Recognition and 3d localization of pedestrian actions from monocular video[C]//2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC). IEEE, 2020: 1-7.
- [10] Gou M, Zhang X, Rates-Borras A, et al. Person re-identification in appearance impaired scenarios[A]. 2016.
- [11] McLaughlin N, Del Rincon J M, Miller P. Recurrent convolutional network for video-based person re-identification[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 1325-1334.
- [12] Varior R R, Shuai B, Lu J, et al. A siamese long short-term memory architecture for human re-identification[C]//European conference on computer vision. Springer, 2016: 135-153.

- [13] Zhao H, Tian M, Sun S, et al. Spindle net: Person re-identification with human body region guided feature decomposition and fusion[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 1077-1085.
- [14] Liu H, Jie Z, Jayashree K, et al. Video-based person re-identification with accumulative motion context[J]. IEEE transactions on circuits and systems for video technology, 2017, 28(10): 2788-2802.
- [15] Baqué P, Fleuret F, Fua P. Deep occlusion reasoning for multi-camera multi-target detection [C]//Proceedings of the IEEE International Conference on Computer Vision. 2017: 271-279.
- [16] Fleuret F, Berclaz J, Lengagne R, et al. Multicamera people tracking with a probabilistic occupancy map[J]. IEEE transactions on pattern analysis and machine intelligence, 2007, 30 (2): 267-282.
- [17] Berclaz J, Fleuret F, Fua P. Multiple object tracking using flow linear programming[C]//2009 Twelfth IEEE international workshop on performance evaluation of tracking and surveillance. IEEE, 2009: 1-8.
- [18] Ge W, Collins R T. Crowd detection with a multiview sampler[C]//European Conference on Computer Vision. Springer, 2010: 324-337.
- [19] Chavdarova T, Fleuret F. Deep multi-camera people detection[C]//2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA). IEEE, 2017: 848-853.
- [20] Dollár P, Wojek C, Schiele B, et al. Pedestrian detection: A benchmark[C]//2009 IEEE conference on computer vision and pattern recognition. IEEE, 2009: 304-311.
- [21] Lima J P, Roberto R, Figueiredo L, et al. Generalizable multi-camera 3d pedestrian detection [C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021: 1232-1240.
- [22] Faster R. Towards real-time object detection with region proposal networks[J]. Advances in neural information processing systems, 2015, 9199(10.5555): 2969239-2969250.
- [23] Hou Y, Zheng L, Gould S. Multiview detection with feature perspective transformation[C]// European Conference on Computer Vision. Springer, 2020: 1-18.
- [24] You Q, Jiang H. Real-time 3d deep multi-camera tracking[A]. 2020.
- [25] Fang H S, Xie S, Tai Y W, et al. RMPE: Regional multi-person pose estimation[C]//ICCV. 2017.
- [26] Li J, Wang C, Zhu H, et al. Crowdpose: Efficient crowded scenes pose estimation and a new benchmark[A]. 2018.
- [27] Xiu Y, Li J, Wang H, et al. Pose Flow: Efficient online pose tracking[C]//BMVC. 2018.

- [28] Jha D, Smedsrud P H, Riegler M A, et al. Resunet++: An advanced architecture for medical image segmentation[C]//2019 IEEE International Symposium on Multimedia (ISM). IEEE, 2019: 225-2255.
- [29] Ristani E, Solera F, Zou R S, et al. Performance measures and a data set for multi-target, multi-camera tracking[C]//ECCV Workshops. 2016.
- [30] Li J, Wang C, Zhu H, et al. Crowdpose: Efficient crowded scenes pose estimation and a new benchmark[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019: 10863-10872.
- [31] Lin T Y, Maire M, Belongie S, et al. Microsoft coco: Common objects in context[C]//European conference on computer vision. Springer, 2014: 740-755.
- [32] López-Cifuentes A, Escudero-Viñolo M, Bescós J, et al. Semantic driven multi-camera pedestrian detection[Z]. 2018.
- [33] Milletari F, Navab N, Ahmadi S A. V-net: Fully convolutional neural networks for volumetric medical image segmentation[J]. 2016 Fourth International Conference on 3D Vision (3DV), 2016.
- [34] Ronneberger O, Fischer P, Brox T. U-net: Convolutional networks for biomedical image segmentation[J]. Springer International Publishing, 2015.
- [35] Zhang Z, Liu Q, Wang Y. Road extraction by deep residual u-net[J]. IEEE Geoscience and Remote Sensing Letters, 2017, PP(99): 1-5.
- [36] Jha D, Smedsrud P H, Riegler M A, et al. Resunet++: An advanced architecture for medical image segmentation[A]. 2019.
- [37] Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015, 39(4): 640-651.
- [38] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 770-778.
- [39] Jie H, Li S, Gang S, et al. Squeeze-and-excitation networks[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, PP(99).
- [40] He K, Zhang X, Ren S, et al. Spatial pyramid pooling in deep convolutional networks for visual recognition[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2014, 37(9): 1904-16.
- [41] Lima J P, Roberto R, Figueiredo L, et al. Generalizable multi-camera 3d pedestrian detection [Z]. 2021.



## 致 谢

衷心感谢导师冯建江副教授和范博昊博士生学长对本人的精心指导，以及舍友们的热情帮助和支持！

特此致谢！



## 声 明

本人郑重声明：所呈交的学位论文，是本人在导师指导下，独立进行研究工作所取得的成果。尽我所知，除文中已经注明引用的内容外，本学位论文的研究成果不包含任何他人享有著作权的内容。对本论文所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确方式标明。

签 名： \_\_\_\_\_ 日 期： \_\_\_\_\_

